Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for ~~facilitating~~ client-server communications ~~internal to a network device~~, comprising:

implementing an interfacing method internal to a cable modem, comprising:

providing the ~~network device~~cable modem with server components and configuring the server components to implement a set of functions;

providing the server components with a set of interface methods for initiating execution of the functions, wherein the set of interface methods correspond to the set of functions;

providing a client component with references to the interface methods, wherein the client component indirectly invokes one of the interface methods by requesting that an interface manager, created by the client component during the initialization of the cable modem, invoke the interface method on behalf of the client component, and wherein the server components, client component, and interface manager execute within the ~~network device~~cable modem;

sending client component requests to the interface manager via a first message queue, wherein each request includes a ~~reference~~ component identifier, an interface identifier, and a function parameter;

processing the client component requests by invoking the interface method corresponding to the interface identifier of the server component corresponding to the ~~reference~~ component identifier included in the request and passing the function parameter to the server component; and

providing a response message regarding the execution of the interface method to the client component via a second message queue, wherein the response message comprises a status.

2. (Previously Presented) The method of claim 1, wherein configuring server components with functionality and providing the interface methods comprises providing a table of pointers for the function.

3. (Previously Presented) The method of claim 2, wherein providing a client component with references to the interface methods comprises providing reference to the table of pointers.

4. (Previously Presented) The method of claim 3, wherein processing client component requests comprises generating requests from the client component for functions from the server components by referencing the table of pointers for the requested functions and generating responses from the server components to provide the functions requested through the table of pointers.

5. (Currently Amended) A method for network device subsystem operations, ~~the method,~~ comprising:

implementing a first component in ~~the~~ a network device, the first component having functions and function pointers corresponding to the functions;

implementing a second component in the network device, the second component having references to the function pointers of the first component, wherein the second component indirectly invokes one of the functions by requesting that an interface manager, created by the second component during the initialization of the network device, invoke the function on behalf of the second component, and wherein the first component, second component, and interface manager execute within the network device;

receiving a request via a message queue from the second component for a function in the first component via a corresponding reference to the function pointer, wherein the request includes a ~~reference~~ component identifier, an interface identifier, and a function parameter; and

generating a response from the first component to provide the requested function result, based on the function parameter, to the interface manager, thereafter the interface manager relaying the result to the second component.

6. (Original) The method of claim 5, wherein implementing the first component comprises providing a table of function pointers.

7. (Previously Presented) The method of claim 6, wherein implementing the second component comprises providing references to the table of function pointers.

8. (Previously Presented) The method of claim 7, wherein receiving a request from the second component comprises referencing the table of pointers for the requested function.

9. (Previously Presented) The method of claim 8, wherein generating a response from the first component comprises receiving the request from the second component, invoking the requested function via the table of pointers, and providing the function parameter to the first component.

10. (Currently Amended) A method of facilitating communications in a network station for a data-over-cable network having a plurality of network stations, the method comprising:

providing a plurality of components in ~~the~~ a network station, each of the plurality of components having a functionality set and a table of pointers for the functionality set;

providing a station manager in the network station having references to the tables of pointers in the plurality of components;

providing an interface manager, created by the station manager during initialization of the network station, for communication via a message queue with ~~the plurality of components and~~ the station manager, wherein the station manager indirectly invokes one of the

component functions by requesting that the interface manager invoke the function on behalf of the station manager, and wherein the components, station manager, and interface manager execute within the network device; and

processing station manager requests for functionality from the plurality of components through the interface manager via the references to the tables of pointers.

11. (Currently Amended) The method of claim 10, wherein processing station manager requests comprises:

generating requests at the station manager for functionality through the references to the tables of pointers and sending the requests for functionality to the interface manager via the message queue, wherein each request includes a ~~reference~~ component identifier, an interface identifier, and a function parameter;

receiving the requests for functionality at the interface manager and invoking the component functionality from the requested functionality sets via the table of pointers; and

providing the function parameter to the component.

12. (Original) The method of claim 11, wherein the requests for functionality are processed serially by the interface manager.

13. (Original) The method of claim 11, wherein the requests for functionality are processed by the interface manager on a first-come first-served basis.

14. (Original) The method of claim 11, wherein providing the plurality of components includes storing data referenced by the pointers are stored in a shared memory area.

15. (Currently Amended) A network device, comprising:

a server component configured with a plurality of functions and function pointers for the plurality of functions and configured to execute within the network device;

a client component configured with references to the function pointers and configured to execute within the network device; and

an interface manager, created by the client component during the initialization of the network device, configured to execute within the network device and configured to receive requests for functions from the client component via a message queue and to invoke the requested functions on behalf of the client component from the server component via the function pointers.

16. (Previously Presented) The device of claim 15, wherein the client component is configured to request server component function execution through the interface manager by providing the reference to the function pointer and a function parameter.

17. (Original) The device of claim 16, wherein the interface manager is configured to receive the references to function pointers and to determine the requested functions to invoke through the references to the function pointers.

18. (Original) The device of claim 15, wherein the server component is configured to include a table of pointers to the functions.

19. (Original) The device of claim 18, wherein the client component is configured to reference the functions through the table of pointers.

20. (Currently Amended) A system for facilitating communications within a network station for a data-over-cable network having a plurality of network stations, the system comprising:

a plurality of components in ~~the~~ a network station, each of the plurality of components having a functionality set and a table of pointers for the functionality set;

a station manager having references to the tables of pointers in the plurality of components; and

an interface manager, created by the station manager during initialization of the network station, for communication via a message queue with ~~the plurality of components and~~ the station manager, the interface manager configured to process station manager requests for functionality from the plurality of components through the interface manager via the references to the tables of pointers, wherein the station manager indirectly invokes one of the component functions by requesting that the interface manager invoke the function on behalf of the station manager, and wherein the components, station manager, and interface manager execute within the network device.

21. (Original) The system of claim 20, wherein the station manager is configured to generate requests for functionality through the references to the tables of pointers and sending the requests for functionality to the interface manager; and the interface manager is configured to receive the requests for functionality and to invoke the functionality from the requested functionality sets via the table of pointers.

22. (Currently Amended) The system of claim 2~~0, wherein the interface manager is configured to process requests for functionality serially.~~1, wherein the network station is a cable modem, the station manager is a cable modem manager, and the interface manager is an OSSI client manager.

23. (Original) The system of claim 20, wherein the interface manager is configured to process requests for functionality on a first-come first-served basis.

24. (Original) The system of claim 20, further comprising a shared memory area for storing all data referenced by the pointers.